

```
/* 03/06/03 by Slava Barsuk */ Source code of program in C  
programming language. This portion of code implements part of  
claim 9 - sending a call to the remote server
```

```
/* power reset client code */  
/* v1.0.0.0 */
```

```
#include <stdio.h> definition of miscellaneous C headers  
#include <unistd.h>  
#include <sys/ioctl.h>  
#include <sys/types.h>  
#include <sys/socket.h>  
#include <netinet/in.h>  
#include <netdb.h>  
#include <string.h>  
#include <strings.h>
```

```
void main(int argc, char *argv[])main body starts here
```

```
{  
    int sock,rc,NB,len; definition of data structures
```

```
    struct sockaddr_in server;  
    struct hostent *hp, *gethostbyname();  
    struct servent *port;
```

```
    char hostname[50];  
    char buf[1024];  
    char wbuf[256];  
    int rbuf[2]={4,12};
```

```
    actual executable code starts here
```

```
        if(getuid()!=0) exit(); check user id of command issuer.  
must be root (superuser), if not, program exits
```

```
        if(argc<2) check that there are at least 2 command line  
arguments, if not print prompt message about command usage and  
exit program
```

```
        { fprintf(stderr,"--- Usage:\n\t%s <hostname>  
[ -r | -h ] \n",argv[0]);  
            exit(4); }
```

```
        len=1024;  
        strcpy(hostname,argv[1]); accept destination server ( remote  
server) hostname  
        port=getservbyname("pwrport",0); resolve tcp communication  
port for further communication
```

```
        sock=socket(AF_INET,SOCK_STREAM,0); create and initialize  
tcp socket structure
```

```
        server.sin_family=AF_INET; set socket type  
        server.sin_len=sizeof(server);  
        hp=gethostbyname(hostname); set destination hostname  
        bcopy(hp->h_addr,&server.sin_addr,hp->h_length);
```

```

server.sin_port=htons(port->s_port); set destination port
rc=connect(sock,(struct sockaddr *)&server,sizeof(server));
- establish connection with remote machine on tcp socket

if(rc!=0) {fprintf(stderr,"--- can't establish connect\n");
exit(4);} check that connection was established successfully ( rc
should be 0), if not display error message and exit program
if(strcmp(argv[2],"-r")==0) analyze remote operation request
- "-r" stands for reboot
{
    fprintf(stderr,"attempt to reboot\n"); display
message that request is for reboot
    rbuf[1]=12; place reboot request code (12) into
buffer
    write(sock, rbuf,8); send message to remote
machine ( write contents of memory referred as rbuf of size 8
bytes into tcp socket sock) claim 9
}
else if(strcmp(argv[2],"-h")==0) if request was not for
reboot "-r", check whether it is for power off(halt) - "-h"
{
    fprintf(stderr,"attempt to halt\n"); display
message that request is for power off(halt)
    rbuf[1]=13; place power off request code(13) into
buffer
    write(sock, rbuf,8); send message to remote
machine( write contents of memory referred as rbuf of size 8
bytes into tcp socket sock) - claim 9
}
close(sock); close connection on tcp socket
}

```

dir_list.txt

total 360

-rw-r-----	1	root	sys	1274	Feb	13	2003	pwrst.c
-rw-r--r--	1	root	sys	1357	Feb	13	2003	pwrhd.c
-rw-r--r--	1	root	sys	3734	Feb	13	2003	pwrhd
-rw-r-----	1	root	sys	1436	Feb	13	2003	pwrn.c
-rwxr-xr-x	1	root	sys	6083	Feb	13	2003	pwrn
-rwxr-xr-x	1	root	sys	5883	Feb	13	2003	pwrst
-rwx-----	1	root	system	288	Feb	13	2003	pwr_up
-rw-r-----	1	root	sys	1620	Feb	13	2003	
pstatus.c								
-rwxr-xr-x	1	root	sys	6179	Feb	13	2003	pstatus
-rwx-----	1	root	system	274	Feb	13	2003	
pwrstatus								
-rwx-----	1	root	system	972	Mar	03	2003	pwrreset
-rw-r-----	1	root	sys	1290	Mar	06	2003	rescl.c
<u>- source file of reset initiator program</u>								
-rwx-----	1	root	system	3983	Mar	06	2003	rescl -
<u>compiled file of reset initiator prog.</u>								
-rw-r--r--	1	root	sys	1738	Mar	31	2003	res.c
<u>-source file "pinned" program</u>								
-rwx-----	1	root	system	4305	Mar	31	2003	res
<u>-compiled file "pinned" program</u>								
-rw-r--r--	1	root	sys	61440	Oct	14	2003	pwr.tar
-rwxr-xr-x	1	root	system	7910	Oct	06	2005	resclx
-rw-r--r--	1	root	system	1857	Oct	06	2005	resx.c
-rwxr-xr-x	1	root	system	11134	Oct	06	2005	resx
-rw-r-----	1	root	system	1433	Oct	28	2005	resclx.c
-rwxr-xr-x	1	root	system	6536	Oct	28	2005	rr
-rw-r--r--	1	root	system	8000	Oct	28	2005	wc